IN THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (Previously Presented) A method for allowing a processor to enter low power states in an information handling system (IHS), the method comprising:

coupling a bus mastering device to a bus mastering device controller via a bus;

coupling the bus mastering device controller to an IHS;

detecting an access request for the bus mastering device, wherein the detecting is performed by a processor of the IHS; and

in response to failing to detect an access request for the bus mastering device within a predetermined period of time, suspending the bus mastering device controller associated with the bus mastering device, wherein the now suspended controller no longer prevents the processor from entering low power states.

2. (Original) The method of claim 1 further comprising:

starting a timer for the predetermined period of time; and

in response to failing to detect an access request for the period of time, expiring the timer, wherein suspending the bus mastering device controller is performed in response to the timer expiring.

3. (Currently Amended) The method of claim 2 further comprising:

in response to detecting an access request for the bus mastering device, restarting the timer for the predetermined period of time.

4. (Currently Amended) The method of claim 3 further comprising:

in response to detecting an access request for the bus mastering device, also resuming operation of the bus mastering device controller if the bus mastering device controller has been suspended.

5. (Original) The method of claim 1 wherein detecting an access request for the bus mastering device includes detecting an input/output request packet (IRP).

6. (Original) The method of claim 4, wherein detecting an access request for the bus mastering device, starting the timer for the predetermined period of time, and resuming the operation of the bus mastering device controller are performed in response to an executable code executable by the IHS.

7. (Original) The method of claim 6, wherein the executable code is a filter driver.

8. (Previously Presented) The method of claim 6 further comprising:

in conjunction with suspending the bus mastering device controller, setting a flag indicating that the bus mastering device controller has been suspended in response to the IHS executing the executable code;

resuming operation of the bus mastering device controller if the flag is set and there is an access request for the bus mastering device; and

clearing the flag.

9. (Original) The method of claim 1, wherein a system management interrupt (SMI) performs suspending the bus mastering device controller.

10. (Original) The method of claim 1, wherein the bus mastering device is a universal serial bus (USB) device and the bus mastering device controller is a USB device controller.

11. (Original) The method of claim 10, wherein the USB device includes a USB storage device.

12. (Original) The method of claim 11, wherein the USB storage device includes a floppy disk drive.

13. (Original) The method of claim 11, wherein the USB storage device includes an optical disk drive.

14. (Previously Presented) An information handling system (IHS) comprising:

a processor capable of entering low power states,

a memory coupled to the processor;

a non-volatile storage, coupled to the processor;

a bus mastering device;

a bus mastering device controller coupled to the bus mastering device via a bus, wherein the bus mastering device controller is coupled to the processor, and wherein the bus mastering device controller is for transferring information between the bus mastering device and the processor; and

an executable code stored in the non-volatile storage for detecting an access request for the bus mastering device and causing the bus mastering device controller to be suspended in response to failing to detect an access request for the bus mastering device within a predetermined period of time, wherein the now suspended controller no longer prevents the processor from entering low power states.

15. (Original) The IHS of claim 14, wherein the executable code starts a timer for the predetermined period of time, expires the timer in response to failing to detect an access request for the bus mastering device within the time period, and wherein causing the bus mastering device controller to be suspended is in response to the timer expiring.

16. (Original) The IHS of claim 15, wherein the executable code restarts the timer for the predetermined period of time in response to detecting an access request for the bus mastering device.

17. (Original) The IHS of claim 16, wherein the executable code further causes the bus mastering device controller to resume operation in response to detecting an access request for the bus mastering device, if the bus mastering device controller has been suspended.

18. (Original) The IHS of claim 14, wherein detecting an access request for the bus mastering device includes detecting an input/output request packet (IRP).

19. (Original) The IHS of claim 14, wherein the process includes a filter driver.

20. (Previously Presented) The IHS of claim 17, wherein the executable code further comprises:

in conjunction with causing the bus mastering device controller to be suspended, setting a flag indicating that the bus mastering device controller has been suspended in response to the IHS executing the executable code;

resuming operation of the bus mastering device controller if the flag is set when there is an access request for the bus mastering device; and

clearing the flag.

21. (Original) The IHS of claim 14, wherein the executable code causes the bus mastering device controller to be suspended by generating a system management interrupt (SMI).

22. (Original) The IHS of claim 14, wherein the bus mastering device is a universal serial bus (USB) device and the bus mastering device controller is a USB device controller.

23. (Original) The IHS of claim 22, wherein the USB device includes a USB storage device.

24. (Original) The IHS of claim 23, wherein the USB storage device includes a floppy disk drive.

25. (Original) The IHS of claim 23, wherein the USB storage device includes an optical disk drive.

26. (Previously Presented) An information handling system (IHS) comprising:

a processor capable of entering low power states,

a non-volatile storage, coupled to the processor;

a bus mastering device;

controller means coupled to the bus mastering device via a bus, wherein the controller means is coupled to the processor, and wherein the controller means is for transferring information between the bus mastering device and the processor and limiting the processor from entering low power states; and

means stored in the non-volatile storage for detecting an access request for the bus mastering device and causing the controller means to be suspended in response to failing to detect an access request for the bus mastering device within a predetermined period of time, wherein the now suspended controller means no longer limits the processor from entering low power states.

27. (Previously Presented) A method for allowing a processor to enter low power states in an information handling system (IHS) comprising:

providing a processor capable of entering low power states,

coupling a non-volatile storage to the processor;

providing a bus mastering device;

coupling a controller means to the bus mastering device via a bus, wherein the controller means is coupled to the processor, and wherein the controller means is for transferring information between the bus mastering device and the processor and limiting the processor from entering low power states; and

storing means in the non-volatile storage for detecting an access request for the bus mastering device and causing the controller means to be suspended in response to failing to detect an access request for the bus mastering device within a predetermined period of time, wherein the now suspended controller means no longer limits the processor from entering low power states.